ABSTRACT

A system and methods for rapid unloading and reorganization of hierarchical databases. Overflow and a window of blocks are progressively read into memory. Unloading proceeds as the scan cylinders window moves ahead. The reading of blocks stays about scan cylinders ahead of the unload. As a segment is unloaded, its space is converted to free IMS space and when appropriate, combined with adjacent free space already in the block. Thus about a window behind the unload point in the data base, all of the segments in a block will have been converted to free space making the block one unit of free space. There will then be no further references to this block and it may be page released back to the OS memory management. Thus no paging subsystem I/O occurs. Where data remains in the data space at the conclusion of the unload, errors are noted that would otherwise have gone unnoticed. When a new database is populated with segments from a disorganized database, the invention provides methods for advance calculation of what the segment RBA is going to be in the database to be reloaded. The proxy dataset consists of proxy blocks. Each proxy block in the proxy data set is represented by a counter that denotes the space available in the proxy block. Segments are unloaded in an algorithmic order that corresponds to a hierarchical relationship in the database. As the segments are unloaded, rather than a literal load of the proxy dataset, the length of each segment is sequentially deducted from the proxy block counter. Because, in a preferred embodiment, the proxy load uses the same algorithm that will be used to actually populate the new reorganized dataset, at each proxy segment load, the counter may be used to calculate the RBA the segment will exhibit in the reorganized dataset. Stored in a table is the segment's current and future RBA. The logical parent or logical child RBA pointer of segments that

participate in logical relationships is used to search the RBA table. When a match is found, the RBA in the segment's prefix is replaced with the corresponding new RBA found in the table. For segments in databases being reorganized, the RBA for segments in logical relationships is used to search the RBA table. When a match is found, the new RBA is placed in the segment pointer field in place of the old or prior RBA.